

Database Prebuffering as a Way to Create a Mobile Control and Information System with Better Response Time

Ondrej Krejcar and Jindrich Cernohorsky

VSB Technical University of Ostrava, Center of Applied Cybernetics, Department of measurement and control, 17. Listopadu 15, 70833 Ostrava Poruba, Czech Republic
{Ondrej.Krejcar, Jindrich.Cernohorsky}@vsb.cz

Abstract. Location-aware services can benefit from indoor location tracking. The widespread adoption of Wi-Fi as the network infrastructure creates the opportunity of deploying WiFi-based location services with no additional hardware costs. Additionally, the ability to let a mobile device determine its location in an indoor environment supports the creation of a new range of mobile control system applications. Main area of interest is in a model of a radio-frequency based system enhancement for locating and tracking users of our control system inside the buildings. The developed framework as it is described here joins the concepts of location and user tracking as an extension for a new control system. The experimental framework prototype uses a WiFi network infrastructure to let a mobile device determine its indoor position. User location is used for data pre-buffering and pushing information from server to user's PDA. All server data is saved as artifacts (together) with its position information in building.

Keywords: prebuffering; localization; framework; Wi-Fi; 802.11x; PDA.

1 Introduction

The usage of various wireless technologies has been increased dramatically and would be growing in the following years. This will lead to the rise of new application domains each with their own specific features and needs. Also these new domains will undoubtedly apply and reuse existing (software) paradigms, components and applications. Today this is easily recognized in the miniaturized applications in network-connected PDAs that provide more or less the same functionality as their desktop application equivalents. It is very likely for these new mobile application domains to adapt new paradigms that specifically target the mobile environment. We believe that an important paradigm is context-awareness. Context is relevant to the mobile user, because in a mobile environment the context is often very dynamic and the user interacts differently with the applications on his mobile device when the context is different. While usually a desktop machine is in a fixed context, a mobile device may be used in work, on the road, during the meeting, or at home. Context is not limited to the physical world around the user, but also incorporates the user's

behavior, terminal and network characteristics. Context-awareness concepts can be found as basic principles in a long-term strategic research for mobile and wireless systems such as formulated in [5]. The majority of context-aware computing to date has been restricted to location-aware computing for mobile applications (location-based services). However, position or location information is a relatively simple form of contextual information. To name a few other indicators of context awareness that make up the parametric context space: identity, spatial information (location, speed), environmental information (temperature), resources that are nearby (accessible devices, hosts), availability of resources (battery, display, network, bandwidth), physiological measurements (blood pressure, heart rate), activity (walking, running), schedules and agenda settings. Context-awareness means that anybody is able to use context information.

We consider location as prime form of context information. We are focused on position determination in an indoor environment. Location information is used to determine an actual user position and his future position. We have performed a number of experiments with the control system, focusing on the position determination we are encouraged by the results. The remainder of this paper describes the conceptual and technical details.

2 Basic Concepts and Technologies of User Localization

The proliferation of mobile computing devices and local-area wireless networks has fostered a growing interest in location-aware systems and services. A key distinguishing feature of such systems is that the application information and/or interface presented to the user is, in general, a function of his/her physical location. The granularity of needed location information could vary from one application to another. For example, locating a nearby printer requires fairly coarse-grained location information whereas locating a book in a library would require fine-grained information. While much research has been focused on a development of services architectures for location-aware systems, less attention has been paid to the fundamental and challenging problem of locating and tracking mobile users, especially in in-building environments. We focus mainly on RF wireless networks in our research. Our goal is to complement the data networking capabilities of RF wireless LANs with accurate user location and tracking capabilities for user needed data pre-buffering. This property we use as an information ground for extension of control system.

2.1 Data Collection

A key step of the proposed research methodology is a data collection phase. We record information about the radio signal as a function of a user's location. The signal information is used to construct and validate models for signal propagation. Among other information, the WaveLAN NIC makes the signal strength (SS) and the signal-to-noise ratio (SNR) available. SS is reported to units of dBm and SNR is expressed in dB. A signal strength of Watts is equivalent to $10 \cdot \log_{10}(s/0.001)$ dBm. For example, signal strength of 1 Watt is equivalent to 30 dBm. Each time the broadcast

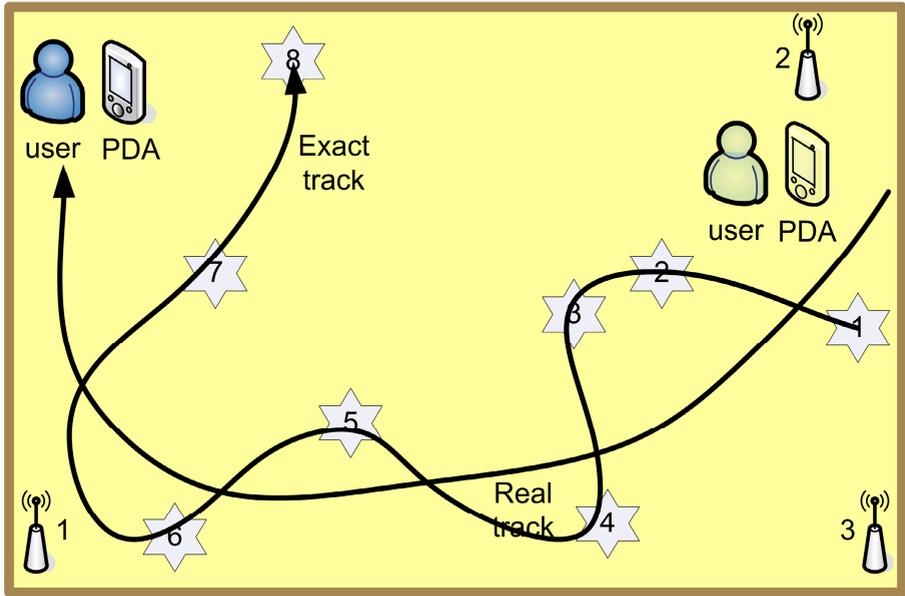


Fig. 1. Localization principle – triangulation

packet is received the WaveLAN driver extracts the SS information from the WaveLAN firmware. Then it makes the information available to user-level applications via system calls. It uses the `wlconfig` utility, which provides a wrapper around the calls to extract the signal information.

2.2 Localization Methodology

The general principle states that if a WiFi-enabled mobile device is close to such a stationary device – Access Point (AP) it may “ask” the provider’s location position by setting up a WiFi connection. If the mobile device knows the position of the stationary device, it also knows that its own position is within a 100-meter range of this location provider. Granularity of location can improve by triangulation of two or several visible WiFi APs. The PDA client will support the application in automatically retrieving location information from nearby location providers, and in interacting with the server. Naturally, this principle can be applied to other wireless technologies. The application (locator) is now implemented in C# using the MS Visual Studio .NET 2005 with .NET compact framework and a special OpenNETCF library enhancement [6].

Schema on figure [Fig. 1] describes a runtime localization process. Each star indicates supposed user location, which was exactly measured and computed. The stars points are exactly measured and computed points of suppose user position. The real track on figure presents real movement of user during the time. The exact track mean computed track from measured WiFi intensity level.

2.3 WiFi Middleware

The WiFi middleware implements the client's side of location determination mechanism on the Windows Mobile 2005 PocketPC operating system and is part of the PDA client application. The libraries used to manage WiFi middleware are: AccessPoint, AccessPointCollection, Adapter, AdapterCollection, AdapterType, ConnectionStatus, Networking, NetworkType, and SignalStrength.

2.4 Predictive Data Push Technology

This part of the project is based on a model of location-aware enhancement, which we have used in created control system. This technique is useful in framework to increase the real dataflow from wireless access point (server side) to PDA (client side). Primary dataflow is enlarged by data pre-buffering. These techniques form the basis of predictive data push technology (PDPT). PDPT copies data from information server to clients PDA to be helpful when user comes at desired location. The benefit of PDPT consists of reduction of time needed to display desired information requested by a user command on PDA. Time delay may vary from a few seconds to number of minutes. It depends on two aspects.

First one is the quality of wireless Wi-Fi connection used by client PDA. A theoretic speed of Wi-Fi connection is max 687 kB/s, because of protocol cost on physical layer (app. 30-40 %). However, the test of transfer rate from server to client's PDA, which we have carried out within our Wi-Fi infrastructure provided the result speed only 80 - 160 kB/s (depends on file size and PDA device).

The second aspect is the size of copied data. Current application records just one set of signal strength measurements at the time (by Locator unit in PDPT Client). By this set of values the actual user position is determined by the PDPT server side. PDPT core responds to location change by selection of the artifact to load to PDPT client buffer. The data transfer speed is widely influenced by the size of these artifacts. For larger artifact size the speed is going down.

Theoretical background and tests were needed to determine an average artifact size. First of all the maximum response time of an application (PDPT Client) for user was needed to be specified. A special book [8] of „Usability Engineering” specified the maximum response time for an application to 10 seconds. During this time the user was focused on the application and was willing to wait for an answer. We used this time period (10 second) to calculate the maximum possible data size of a file transferred from server to client (during this period). If transfer speed was from 80 to 160 kB/s the result file size was from 800 to 1600 kB. The next step was an average artifact size definition. We used a sample database of network architecture building plan (Autocad file type), which contained 100 files of average size of 470 kB. The client application can download during the 10 second period from 2 to 3 artifacts. The problem is the time, which is needed for displaying them. In case of Autocad file type we measured this time to average 45 seconds. This time consumption is certainly not acceptable, for this reason we are looking for a better solution. We need to use some basic data format, which can be displayed by PDA natively (BMP, JPG, GIF) without any additional striking time consumption. The solution is in format conversion from any to this native (for PDA devices). In case of sound and video format we also recommend using basic data format (wav, mp3, wmv, mpg).

The final result of our real tests and consequential calculations is definition of artifact size to average value of 500 kB. The buffer size may differ from 50 to 100 MB in case of 100 to 200 artifacts.

2.5 PDPT Framework Data Artifact Management

The PDPT Server SQL database manages the information (for example data about Ethernet hardware such as Ethernet switch, UTP socket, CAT5 cable lead, etc.) in the context of their location in building environment. This context information is same as location information about user track. The PDPT core controls data, which are copied from the server to PDA client by context information (position info). Each database artifacts must be saved in database along the position information, to which it belongs.

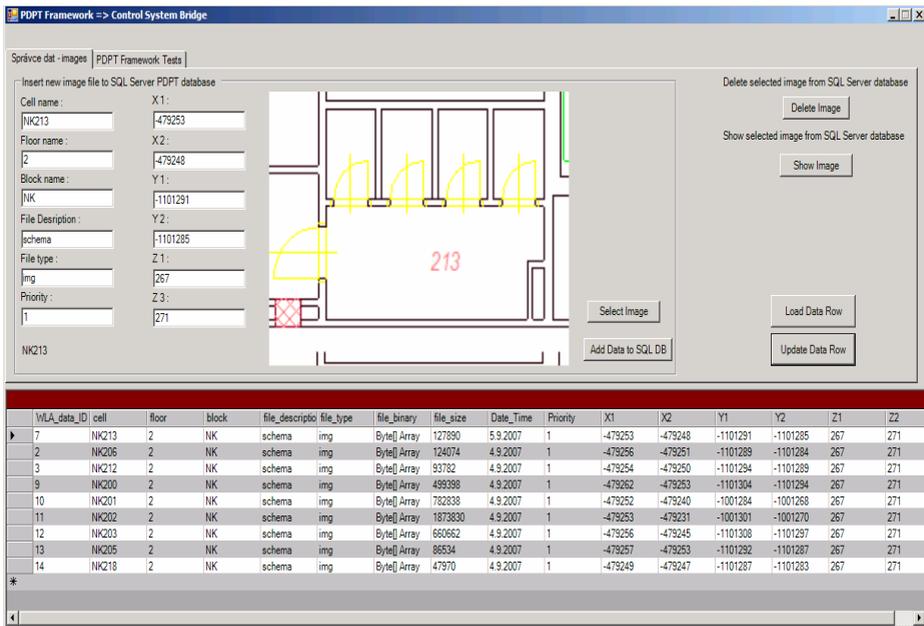


Fig. 2. PDPT Framework data artifact management

During the creating process of PDPT Framework the new software application called “Data Artifacts Manager” was created. This application manages the artifacts in WLA database (localization oriented database). User can set the priority, location, and other metadata of the artifact. This manager substitutes the online conversion mechanism, which can transform the real online control system data to WLA database data artifacts during the test phase of the project. This manager can be also used in case of offline version of PDPT Framework usage. The artifacts manager in this offline case is shown at [Fig 2].

The Manager allows to the administrator to create a new artifact from multimedia file (image, video, sound, etc.), and edit or delete the existing artifact. The left side of the screen contains the text field of artifact metadata as a position in 3D space. This position is determined by artifact size (in case of building plan) or binding of artifact to some part of a building in 3D space. The 3D axis is possible to take from building plan by some GIS software like Quantum GIS or by own implementation [7]. The central part represents a multimedia file and right side contains the buttons to create, edit, or delete the artifact. The lower part of the application screen shows actual artifacts in WLA database located on SQL Server.

2.6 Framework Design

PDPT framework design is based on the most commonly used server-client architecture. To process data the server has online connection to the control system. Technology data are continually saved to SQL Server database [3] and [1].

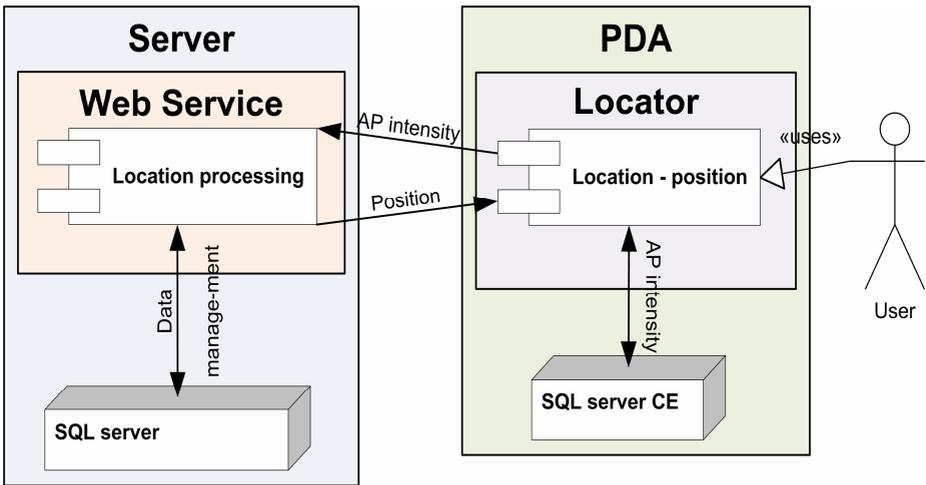


Fig. 3. System architecture – UML design

The part of this database (desired by user location or his demand) is replicated online to client’s PDA, where it is visualized on the screen. User PDA has location sensor component, which continuously sends the information about nearby AP’s intensity to the framework kernel. The kernel processes this information and makes a decision if or how the part of SQL Server database will be replicated to client’s SQL Server CE database. The kernel decisions constitute the most important part of whole framework, because the kernel must continually compute the position of the user and track, and make a prediction of his future movement. After doing this prediction the appropriate data (part of SQL Server database) are pre-buffered to client’s database for the future possible requirements. The PDPT framework server is created as Microsoft web service to handle a bridge between SQL Server and PDPT PDA Clients.

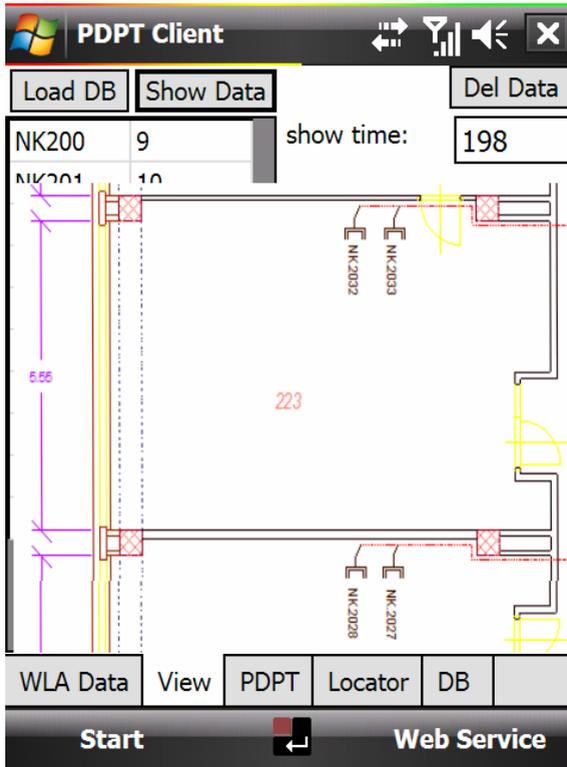


Fig. 4. PDPT Client – Windows Mobile application

2.7 PDPT Client

For testing and tuning of PDPT Core was created the PDPT Client application. This client realizes classical client to the server side and an extension by PDPT and Locator module. Figure [Fig. 4] shows classical view of data presentation from MS SQL CE database to user (in this case the image of Ethernet network in company area plan). Each process running in a PDPT client is measured in millisecond resolution to provide a feedback from real situation. Tabs PDPT and Locator present a way to tune the settings of PDPT values.

3 Experiments

We have executed a number of indoor experiments with the PDPT framework using the PDPT PDA application. WiFi access points are placed at different locations in building, where the access point cells partly overlap. We have used triangulation principle of AP intensity to obtain a better granularity. It has been found that the location determination mechanism selects the access point that is the closest to the mobile user as the best location provider. Also after the loss of IP connectivity,

switching from one access point to another (a new best location provider) takes place within a second in the majority of cases, resulting in only temporary loss of IP connectivity. This technique partially uses a special Radius server [4] to realize “roaming” known in cell’s networks. User, who lost the existing signal of AP is required to ask the new AP to receive IP. This is known as “renew” in Ethernet networks. At the end of this process, user has his identical old IP and connection to new AP. Other best technique to realize roaming is using of WDS (Wireless Decision System). Currently, the usability of the PDPT PDA application is somewhat limited due to the fact that a device has to be continuously powered. If it is not, the WiFi interface and the application cannot execute the location determination algorithm and the PDPT server does not receive location updates from the PDA.

3.1 Data Transfer Increase Tests Using PDPT Framework

The main result of utilization of PDPT framework is reduction of data transfer speed. The test is focused on the real usage of developed PDPT Framework and its main issue at increased data transfer.

Table 1. Data transfer tests description

Test	Type	Mode	Data	Time	Speed
1	HTC Blueangel	SQL CE	2949	5	643
2		SQL CE	4782	9	2228
3		SQL	2949	34	80
4		SQL	4782	57	69
5		PDPT	2949	12	234
6		PDPT	4782	20	278
7	HTC Universal	SQL CE	2949	3	514
8		SQL CE	4782	6	1782
9		SQL	2949	21	51
10		SQL	4782	38	64
11		PDPT	2949	9	214
12		PDPT	4782	16	2228

At the Table 1 is a summary of eighteen tests with three types of PDA and three types of data transfer mode. Each of these eighteen tests are fivefold reiterated for a better accuracy. Data in the table are average values from each iterations.

The mode column may contain three different data transfer modes. The SQL CE mode represents data saved at mobile device memory (SQL Server CE) and a data transfer time is very high.

The second mode SQL shows data, which are stored at server (SQL Server 2005). Primary data are loaded over Ethernet / Internet to SQL Server CE of mobile device and secondary data are shown to user. Data transfers time consumption of this method is generally very high, which results in a high waiting time for users.

The third data mode PDPT is a combination of previous two methods. The PDPT mode provides very good results in the form of a data transfer acceleration. A realization of this test consists of user movement from a sample location A to B at three different way directions. Location B was a destination with requested data,

which is not contained at SQL CE buffer in mobile device before test. The result time of this mode consist of average time from view a collection of requested data artifacts on client PDA. If the requested artifact is in SQL CE buffer before request, the time is very short. If the artifact is not present in buffer, PDPT client must download them from server. Final time for third method represents the real usage result.

Acknowledgment. This work was supported by the Ministry of Education of the Czech Republic under Project 1M0587

4 Conclusions

The main objective of this paper is in the enhancement of control system for locating and tracking of users inside a building. It is possible to locate and track the users with high degree of accuracy. In this paper we have presented the control system framework that uses and handles location information and control system functionality. The indoor location of a mobile user is obtained through an infrastructure of WiFi access points. This mechanism measures the quality of the link of nearby location provider access points to determine actual user position. User location is used in the core of server application of PDPT framework to data pre-buffering and pushing information from server to user's PDA. Data pre-buffering is the most important technique to reduce time from user request to system response. The experiments show that the location determination mechanism provides a good indication of the actual location of the user in most cases. The median resolution of the system is approximately five meters. Some inaccuracy does not influence the way of how the localization is derived from the WiFi infrastructure. For the PDPT framework application this was not found to be a big limitation for the PDPT framework application as it can be found at chapter Experiments. The experiments also show that the current state of the basic technology, which was used for the framework (mobile device hardware, PDA operating system, wireless network technology) is now at the level of a high usability of the PDPT application [13].

References

1. Reynolds, J.: *Going Wi-Fi: A Practical Guide to Planning and Building an 802.11 Network*. CMP Books (2003)
2. Wigley, A., Roxburgh, P.: *ASP.NET applications for Mobile Devices*. Microsoft Press, Redmond (2003)
3. Tiffany, R.: *SQL Server CE Database Development with the .NET Compact Framework*. Apress (2003)
4. The Internet Engineering Task Force RADIUS Working Group, <http://www.ietf.org/>
5. The Wireless World Research Forum (WWRF), <http://www.wireless-world-research.org/>
6. OpenNETCF - Smart Device Framework, <http://www.opennetcf.org>
7. Horak, J., Unucka, J., Stromsky, J., Marsik, V., Orlik, A.: TRANSCAT DSS architecture and modelling services. *Journal: Control and Cybernetics* 35, 47–71 (2006)
8. Nielsen, J.: *Usability Engineering*. Morgan Kaufmann, San Francisco (1994)

9. Krejcar, O.: Prebuffering as a way to exceed the data transfer speed limits in mobile control systems. In: *Icinco 2008, 5th International Conference on Informatics in Control, Automation and Robotics*. Insticc Press, Funchal (2008)
10. Evennou, F., Marx, F.: Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning. *Eurasip journal on applied signal processing*, Hindawi publishing corp., New York (2006)
11. Olivera, V., Plaza, J., Serrano, O.: WiFi localization methods for autonomous robots. *Journal Robotica* 24, 455–461 (2006)
12. Salazar, A.: Positioning Bluetooth (R) and Wi-Fi (TM) systems. *Journal IEEE transactions on consumer electronics* 50, 151–157 (2004)
13. Janckulik, D., Krejcar, O., Martinovic, J.: Personal Telemetric System – Guardian. In: *Biodevices 2008*, pp. 170–173. Insticc Setubal, Funchal (2008)